

# Specyfikacja wymagań systemowych

3@KASK

21 kwietnia 2009

Symbol projektu: 3@KASK	Opiekun projektu: Tomasz Boiński
Nazwa Projektu: Wizualizacja grafów za pomocą biblioteki Prefuse	

Nazwa Dokumentu: Specyfikacja wymagań systemowych	Nr wersji: 0.0
Odpowiedzialny za dokument: Piotr Orłowski	Data pierwszego sporządzenia: 15 kwietnia 2009
Przeznaczenie: ???	Data ostatniej aktualizacji: 21 kwietnia 2009

## Historia dokumentu

Wersja	Opis modyfikacji	Rozdział/strona	Autor modyfikacji	Data
1	Stworzenie	wszystkie	Grupa projektowa	15.04.09
2	Wpisanie celów			

## Spis treści

<b>1</b>	<b>Cele systemu</b>	<b>3</b>
1.1	Cele biznesowe . . . . .	3
1.2	Cele funkcjonalne . . . . .	3
<b>2</b>	<b>Otoczenie systemu</b>	<b>4</b>
2.1	Użytkownicy . . . . .	4
2.2	Systemy zewnętrzne . . . . .	4
<b>3</b>	<b>Przewidywane komponenty systemu</b>	<b>4</b>
3.1	Podsystemy . . . . .	4
3.2	Komponenty sprzętowe . . . . .	4
3.3	Programowe . . . . .	4
<b>4</b>	<b>Wymagania funkcjonalne</b>	<b>4</b>
<b>5</b>	<b>Wymagania na dane</b>	<b>5</b>
<b>6</b>	<b>Wymagania jakościowe</b>	<b>5</b>
6.1	Wymagania w zakresie wiarygodności . . . . .	5
6.2	Wymagania w zakresie wydajności . . . . .	6
6.3	Wymagania w zakresie elastyczności . . . . .	6
6.4	Wymagania w zakresie użyteczności . . . . .	6
<b>7</b>	<b>Sytuacje wyjątkowe</b>	<b>6</b>
<b>8</b>	<b>Dodatkowe wymagania</b>	<b>6</b>
8.1	Wymagania sprzętowe . . . . .	6
8.2	Wymagania programowe . . . . .	7
8.3	Inne wymagania . . . . .	7
<b>9</b>	<b>Kryteria akceptacyjne</b>	<b>7</b>
	<b>Literatura</b>	<b>8</b>

# 1 Cele systemu

## 1.1 Cele biznesowe

Cele biznesowe precyzują korzyści związane z wdrożeniem systemu.

CB001	Ułatwienie pracy programistom tworzącym aplikacje wizualizujące ontologie
Opis:	Istnieje zapotrzebowanie na bibliotekę tłumaczącą OWL bezpośrednio na elementy graficzne.
Źródło:	Wstępna specyfikacja projektu
Priorytet:	bardzo ważne

CB002	Ułatwienie zakończenia projektu OCS
Opis:	Moduł wizualizujący ontologie w OCS wymaga modernizacji i rozbudowy
Źródło: Klient - mgr inż. Tomasz Boiński	
Priorytet:	bardzo ważne

CB003	Zwiększenie atrakcyjności portalu OCS
Opis:	Poprawa estetyki modułu wizualizującego ontologię może przyczynić się do sukcesu portalu po jego wdrożeniu
Źródło:	Klient - mgr inż. Tomasz Boiński
Priorytet:	mało ważne

CB004	
Opis:	
Źródło:	
Priorytet:	

## 1.2 Cele funkcjonalne

Cele funkcjonalne wymieniają główne funkcje, które ma spełniać system.

CF001	Intuicyjne API
Opis:	
Źródło:	
Priorytet:	średnio ważne

CF002	Dobra dokumentacja
Opis:	Przygotowanie dokumentacji w Javadoc ułatwi pracę użytkownikom biblioteki.
Źródło:	Klient - mgr inż. Tomasz Boiński
Priorytet:	bardzo ważne

CF003	Wizualizacja ontologii
Opis:	Stworzenie biblioteki, która pozwoli na wizualizację obiektów OWL API przy użyciu odpowiedniej biblioteki
Źródło:	Specyfikacja projektu
Priorytet:	bardzo ważne

CF004	Umożliwienie graficznej edycji i dodawania obiektów OWL API
Opis:	Dostarczenie tej funkcjonalności ułatwi tworzenie programów z interfejsem pozwalającym na edycję ontologii
Źródło:	Klient - mgr inż. Tomasz Boiński i Anna Jaworska
Priorytet:	średnio ważne

CF005	Udostępnienie informacji do debuggowania
Opis:	Biblioteka powinna wysyłać komunikaty informacyjne, ostrzegawcze oraz informujące o błędach na strumieniu
Źródło:	Standard tworzenia biblioteki [1]
Priorytet:	średnio ważne

## 2 Otoczenie systemu

Zespół projektowy musi poznać otoczenie, w jakim ma pracować system. Z rozmów z klientem powinno dać się wyszczególnić użytkowników oraz systemy zewnętrzne. Jeśli się nie da, to otoczenie systemu trzeba będzie zdefiniować w trakcie analizy funkcjonalnej.

### 2.1 Użytkownicy

Tutaj jest ID	A tutaj nazwa
Opis:	
Potrzeby:	
Zadania:	
Źródło:	
Priorytet:	

### 2.2 Systemy zewnętrzne

Tutaj jest ID	A tutaj nazwa
Opis:	
Interfejsy:	
Źródło:	
Priorytet:	

## 3 Przewidywane komponenty systemu

Wyszczególnienie komponentów systemu powinno pomóc w uzyskaniu kompletności wymagań. Trzeba wówczas sprawdzić, czy każdy komponent ma jakieś wymagania (zwłaszcza funkcjonalne). W przypadku bardziej złożonego systemu może być konieczne wyszczególnienie podsystemów.

### 3.1 Podsystemy

Specyfika projektu sprawia, że podsystemy nie będą rozpatrywane.

### 3.2 Komponenty sprzętowe

Specyfika projektu sprawia, że komponenty sprzętowe nie będą rozpatrywane.

### 3.3 Programowe

Tutaj jest ID	A tutaj nazwa
Opis:	
Powiązania:	
Źródło:	
Priorytet:	

## 4 Wymagania funkcjonalne

Wymagania funkcjonalne stanowią mocno rozbudowaną część specyfikacji. Można je podzielić na grupy dotyczące różnych zadań, różnych użytkowników (systemów zewnętrznych) albo różnych komponentów.

WF001	Udostępnienie kilku algorytmów wizualizacji
Opis:	Biblioteka powinna udostępniać kilka trybów prezentacji grafów (np. w formie drzewa, w formie gwiazdy)
Dotyczy:	
Źródło:	
Powiązania:	
Priorytet:	

WF002	Parametryzacja trybów wizualizacyjnych
Opis:	Domyślne parametry w trybach wizualizacji (takie jak długość krawędzi grafu, automatyczne układanie)
Dotyczy:	
Źródło:	
Powiązania:	
Priorytet:	

WF003	
Opis:	
Dotyczy:	
Źródło:	
Powiązania:	
Priorytet:	
WF002	
Opis:	
Dotyczy:	
Źródło:	
Powiązania:	
Priorytet:	
WF002	
Opis:	
Dotyczy:	
Źródło:	
Powiązania:	
Priorytet:	

## 5 Wymagania na dane

Wymagania na dane pomagają w określeniu, jakie dane będą przetwarzane w systemie. Nie trzeba precyzować wszystkich danych. Szczegóły znajdują się w projekcie bazy danych.

WD001	Obsługa obiektów OWL API
Opis:	Bibliotek będzie przystosowana do pobierania, obróki i zwracania obiektów OWL API
Powiązania:	
Źródło:	Klient - mgr inż. Tomasz Boiński
Priorytet:	bardzo ważne

## 6 Wymagania jakościowe

Określenie wymagań jakościowych ułatwia późniejsze uzyskanie wysokiej jakości systemu. Podział wymagań jakościowych na kategorie jest związany z drzewem jakości (dotyczy wszystkich gałęzi drzewa za wyjątkiem funkcjonalności).

### 6.1 Wymagania w zakresie wiarygodności

Wymagania w zakresie wiarygodności będą rozszerzały wymagania funkcjonalne.

Tutaj jest ID	A tutaj nazwa
Opis:	
Powiązania:	
Źródło:	
Priorytet:	

## 6.2 Wymagania w zakresie wydajności

Wymagania w zakresie wydajności będą miały zastosowanie w czasie projektowania architektury systemu.

Tutaj jest ID	A tutaj nazwa
Opis:	
Powiązania:	
Źródło:	
Priorytet:	

## 6.3 Wymagania w zakresie elastyczności

Wymagania w zakresie elastyczności będą miały zastosowanie w czasie wyboru koncepcji systemu.

Tutaj jest ID	A tutaj nazwa
Opis:	
Powiązania:	
Źródło:	
Priorytet:	

## 6.4 Wymagania w zakresie użyteczności

Wymagania w zakresie użyteczności będą brane pod uwagę głównie w czasie projektowania interfejsu użytkownika.

Tutaj jest ID	A tutaj nazwa
Opis:	
Powiązania:	
Źródło:	
Priorytet:	

## 7 Sytuacje wyjątkowe

Sytuacje wyjątkowe stanowią dalsze rozszerzenie wymagań funkcjonalnych i wiarygodnościowych.

Tutaj jest ID	A tutaj nazwa
Opis:	
Powiązania:	
Źródło:	
Priorytet:	

## 8 Dodatkowe wymagania

W tym miejscu podaje się te wymagania, które nie mieszczą się w zakresie poprzednich kategorii wymagań.

### 8.1 Wymagania sprzętowe

Wymagania sprzętowe można by umieścić w ramach specyfikacji komponentów sprzętowych, ale jeśli jest wiele komponentów sprzętowych różnych z punktu widzenia funkcjonalnego, ale o wspólnych wymaganiach sprzętowych, to można te wymagania umieścić właśnie tutaj.

Tutaj jest ID	A tutaj nazwa
Opis:	
Dotyczy:	
Źródło:	
Priorytet:	

## 8.2 Wymagania programowe

Trzeba odróżniać rzeczywiste wymagania programowe klienta od jego sugestii (np. przez podanie opcjonalnego priorytetu).

Tutaj jest ID	A tutaj nazwa
Opis:	
Dotyczy:	
Źródło:	
Priorytet:	

## 8.3 Inne wymagania

Tutaj jest ID	A tutaj nazwa
Opis:	
Dotyczy:	
Źródło:	
Priorytet:	

## 9 Kryteria akceptacyjne

Tu podać kryteria, jakim zostanie poddany gotowy system przed ostatecznym jego przyjęciem.

Tutaj jest ID	A tutaj nazwa
Opis:	
Dotyczy:	
Źródło:	
Priorytet:	

## **Literatura**

- [1] Greg Travis. Build your own java library. publikacja elektroniczna.  
[http://www.digilife.be/quickreferences/PT/Build your own Java library.pdf](http://www.digilife.be/quickreferences/PT/Build%20your%20own%20Java%20library.pdf).